UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/780,264	02/16/2004	Mario I. Wolczko	188073/US	2218

66083 7590 04/28/2010

Dorsey & Whitney LLP
on behalf of Sun Microsystems, Inc.
370 Seventeenth Street
Republic Plaza Building, Suite 4700
Denver, CO 80202-5647

EXAMINER	
YIGDALL, MICHAEL J	

ART UNIT	PAPER NUMBER
2192	

NOTIFICATION DATE	DELIVERY MODE
04/28/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

Docketing-DV@dorsey.com
DocketingDV@dorsey.foundationip.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/780,264
Filing Date: February 16, 2004
Appellant(s): WOLCZKO ET AL.

Gregory P. Durbin
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed on February 4, 2010 appealing from the final Office action mailed on August 4, 2009.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

1-18 and 20-23.

(4) Status of Amendments after Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to Be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the

appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

6,000,044	CHRYSOS et al.	12-1999
7,448,025	KALAFATIS et al.	11-2008

(9) Grounds of Rejection

The following ground(s) of rejection, set forth in the final Office action mailed on August 4, 2009, are applicable to the appealed claims:

- Claims 1-18 and 20-23 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,000,044 to Chrysos et al. ("Chrysos") in view of U.S. Patent No. 7,448,025 to Kalafatis et al. ("Kalafatis").

Claim 1

Chrysos teaches a method of sampling instructions executing in a multi-threaded processor (see, for example, the abstract) comprising:

selecting an instruction for sampling (see, for example, column 10, lines 19-25, which shows selecting an instruction for sampling);

storing sampling information relating to the instruction (see, for example, column 11, lines 31-38, which shows storing such sampling information);

determining whether the sampling information includes an event of interest to a particular thread within which the instruction is executing (see, for example, column 15, lines 32-42, which shows filtering the sampling information for events of interest, and column 12, lines 1-4, which further shows filtering the sampling information based on the thread of execution).

Chrysos further teaches reporting the sampling information when the sampling information includes an event of interest (see, for example, FIG. 7B and column 17, lines 34-61), but does not explicitly describe:

reporting the sampling information to the particular thread when the sampling information includes an event of interest.

Nonetheless, one of ordinary skill in the art could, with predictable results, implement the teachings of Chrysos such that the sampling information is reported to the particular thread when the sampling information includes an event of interest to the particular thread.

For example, in an analogous art, Kalafatis teaches qualifying events of interest in a multi-threaded processor based on the particular thread that executes the instructions (see, for example, column 2, lines 39-55). Kalafatis describes that such qualification provides versatility in reporting the event information (see, for example, column 6, lines 41-58).

Therefore, in view of Kalafatis, it would have been obvious to those of ordinary skill in the art at the time the invention was made to implement the teachings of Chrysos so as to report

the sampling information to the particular thread when the sampling information includes an event of interest.

Claim 2

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing a register with a bit vector representing a plurality of events of interest; and

wherein the determining whether the sampling information includes the event of interest further includes comparing the sampling information relating to the instruction to the bit vector (see, for example, column 16, lines 52-55).

Claim 3

The rejection of claim 2 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the comparing is via at least one of a mask operation or a more expressive operation (see, for example, column 16, lines 52-55).

Claim 4

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the selecting the instruction is without regard to a thread to which the instruction is bound (see, for example, column 6, lines 48-49).

Claim 5

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests identifying a thread to which the instruction is bound when the instruction is selected (see, for example, column 14, lines 53-64).

Claim 6

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing filtering criteria on a per-thread basis (see, for example, column 15, lines 21-43).

Claim 7

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing a single set of filtering criteria; and, scheduling sampling among a plurality of threads via software (see, for example, column 15, lines 21-43).

Claim 8

Chrysos teaches a method of sampling instructions executing in a multi-threaded processor (see, for example, the abstract) comprising:

setting a candidate counter to a number (see, for example, column 14, lines 40-52, which shows setting a counter to a value);

selecting an instruction for sampling (see, for example, column 10, lines 19-25, which shows selecting an instruction for sampling);

storing information relating to the instruction (see, for example, column 11, lines 31-38, which shows storing such information);

determining whether all events for the instruction have occurred (see, for example, column 15, lines 43-49, which shows determining that all events for the instruction are complete).

Chrysos further teaches decrementing the counter (see, for example, column 14, line 64 to column 15, line 4) and filtering the information based on the thread of execution (see, for example, column 12, lines 1-4), but does not explicitly describe:

decrementing the candidate counter when all events for the instruction have occurred and when the instruction corresponds to a desired sampled thread.

Nonetheless, one of ordinary skill in the art could, with predictable results, implement the teachings of Chrysos such that the candidate counter is decremented when all events for the instruction have occurred and when the instruction corresponds to a desired sampled thread.

For example, in an analogous art, Kalafatis teaches qualifying events of interest in a multi-threaded processor based on the particular thread that executes the instructions (see, for example, column 2, lines 39-55). Kalafatis describes that such qualification provides versatility in reporting the event information (see, for example, column 6, lines 41-58).

Therefore, in view of Kalafatis, it would have been obvious to those of ordinary skill in the art at the time the invention was made to implement the teachings of Chrysos so as to decrement the candidate counter when all events for the instruction have occurred and when the instruction corresponds to a desired sampled thread.

Chrysos in view of Kalafatis further teaches or suggests:

determining whether the candidate counter equals zero (see, for example, column 14, line 64 to column 15, line 4, which shows determining that the counter underflows or overflows); and

reporting the instruction to a particular thread when the candidate counter equals zero (see, for example, FIG. 7B and column 17, lines 34-61, which shows reporting the information).

Claim 9

The rejection of claim 8 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the information relating to the instruction represents an instruction history (see for example column 6, lines 48-49), and the instruction history includes information relating to at least one of an events value, a program counter value, a branch target address value, an effective memory address value, a latency value, a number in issue bundle value, a number in retire bundle value, a privilege value, a branch history value and a number in fetch bundle value (see, for example, column 6, lines 48-49).

Claim 10

The rejection of claim 8 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the selecting the instruction is without regard to a thread to which the instruction is bound (see, for example, column 14, lines 53-64).

Claim 11

The rejection of claim 8 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests identifying a thread to which the instruction is bound when the instruction is selected (see, for example, column 14, lines 53-64).

Claim 12

The rejection of claim 8 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing filtering criteria on a per-thread basis (see, for example, column 15, lines 21-34).

Claim 13

The rejection of claim 8 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing a single set of filtering criteria; and, scheduling sampling among a plurality of threads via software (see, for example, column 15, lines 21-34).

Claim 14

Chrysos teaches a method of sampling instructions executing in a multi-threaded processor comprising:

setting a candidate counter to a number (see, for example, column 14, lines 40-52, which shows setting a counter to a value);

selecting an instruction for sampling (see, for example, column 10, lines 19-25, which shows selecting an instruction for sampling);

storing information relating to the instruction(see, for example, column 11, lines 31-38, which shows storing such information);

determining whether all events for the instruction have occurred (see, for example, column 15, lines 43-49, which shows determining that all events for the instruction are complete).

Chrysos further teaches filtering the information for events of interest (see, for example, column 15, lines 32-42) and filtering the information based on the thread of execution (see, for example, column 12, lines 1-4), but does not explicitly describe:

determining whether the instruction includes events of interest, the events of interest including whether the instruction corresponds to a desired sampled thread.

Nonetheless, one of ordinary skill in the art could, with predictable results, implement the teachings of Chrysos so as to determine whether the instruction includes events of interest, the events of interest including whether the instruction corresponds to a desired sampled thread.

For example, in an analogous art, Kalafatis teaches qualifying events of interest in a multi-threaded processor based on the particular thread that executes the instructions (see, for example, column 2, lines 39-55). Kalafatis describes that such qualification provides versatility in reporting the event information (see, for example, column 6, lines 41-58).

Therefore, in view of Kalafatis, it would have been obvious to those of ordinary skill in the art at the time the invention was made to implement the teachings of Chrysos so as to determine whether the instruction includes events of interest, the events of interest including whether the instruction corresponds to a desired sampled thread.

Chrysos in view of Kalafatis further teaches or suggests:

decrementing the candidate counter when all events for the instruction have occurred and when the instruction includes events of interest (see, for example, column 14, line 64 to column 15, line 4, which shows decrementing the counter);

determining whether the candidate counter equals zero (see, for example, column 14, line 64 to column 15, line 4, which shows determining that the counter underflows or overflows); and

reporting the instruction to a particular thread when the candidate counter equals zero (see, for example, FIG. 7B and column 17, lines 34-61, which shows reporting the information).

Claim 15

The rejection of claim 14 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests providing a register with a bit vector representing events of interest; and

wherein the determining whether the instruction includes events of interest further includes comparing the information relating to the instruction to the bit vector (see, for example, column 16, lines 52-55).

Claim 16

The rejection of claim 14 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the information relating to the instruction represents an instruction history, and the instruction history includes information relating to at least one of an event value, a program counter value, a branch target address value, an effective memory address value, a latency value, a number in issue bundle value, a number in retire bundle value, a privileged value, a branch history value and a number in fetch bundle value (see, for example, column 6, lines 48-49).

Claim 17

The rejection of claim 14 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the selecting an instruction for sampling is based upon sample selection criteria; and

the sample selection criteria include information relating to a desired sampled thread (see, for example, column 15, lines 30-35).

Claim 18

Chrysos teaches a multi-threaded processor (see, for example, FIG. 1) comprising:

a sampling logic configured to determine whether an instruction executed in the processor corresponds to a desired sampled thread (see, for example, column 10, lines 19-25, which shows determining that an instruction is selected for sampling);

a sampling register logic coupled to the sampling logic (see, for example, FIG. 2B);

an instruction history register logic coupled to the sampling register logic, the instruction history register logic storing information relating to the instruction (see, for example, column 11, lines 31-38, which shows storing such information);

a sample filtering and counting logic coupled to the sampling logic (see, for example, column 15, lines 32-42, which shows filtering, and column 14, line 64 to column 15, line 4, which shows counting).

Chrysos further teaches filtering the information based on the thread of execution (see, for example, column 12, lines 1-4), but does not explicitly describe:

wherein the sample filtering and counting logic is replicated on a per thread basis.

Nonetheless, one of ordinary skill in the art could, with predictable results, implement the teachings of Chrysos such that the sample filtering and counting logic is replicated on a per thread basis.

For example, in an analogous art, Kalafatis teaches qualifying events of interest in a multi-threaded processor based on the particular thread that executes the instructions (see, for example, column 2, lines 39-55). Kalafatis describes that such qualification provides versatility in reporting the event information (see, for example, column 6, lines 41-58).

Therefore, in view of Kalafatis, it would have been obvious to those of ordinary skill in the art at the time the invention was made to implement the teachings of Chrysos such that the sample filtering and counting logic is replicated on a per thread basis.

Chrysos in view of Kalafatis further teaches or suggests:

a notification logic, the notification logic reporting to a particular thread the information relating to the instruction if the instruction corresponds to the desired sampled thread (see, for example, column 6, lines 60-65, and see, for example, FIG. 7B and column 17, lines 34-61, which shows reporting the information).

Claim 20

The rejection of claim 18 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the sampling register logic includes a register with a bit vector representing events of interest; and

wherein the sampling logic determines whether the instruction includes events of interest by comparing the information relating to the instruction to the bit vector (see, for example, column 16, lines 52-55).

Claim 21

The rejection of claim 18 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the information relating to the instruction represents an instruction history (see, for example, column 6, lines 40-45), and the instruction history includes information relating to at least one of an events value, a program counter value, a branch target address value, an effective memory address value, a latency value, a number in issue bundle

value, a number in retire bundle value, a privileged value, a branch history value and a number in fetch bundle value (see, for example, column 6, lines 48-49).

Claim 22

The rejection of claim 18 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that the sampling register logic includes a sample selection criteria register storing sample selection criteria (see, for example, column 16, lines 52-55); and the sample selection criteria include information relating to a desired sampled thread (see, for example, column 15, lines 30-35).

Claim 23

The rejection of claim 1 is incorporated, and Chrysos in view of Kalafatis further teaches or suggests that storing the sampling information further comprises storing the sampling information to a memory register shared by multiple threads (see, for example, column 11, lines 31-38, which shows storing the sampling information to a shared register file in memory).

(10) Response to Arguments

Appellant contends that Chrysos does not disclose reporting "the sampling information" or "the instruction" to a particular thread (brief, page 8). Specifically, Appellant contends that Chrysos "discloses sampling and storing, but fails to disclose reporting any stored information, let alone reporting to a particular thread" (brief, page 9). Appellant contends that the apparatus of Chrysos "only stores the gathered information and does not disclose reporting the information to a particular thread" (brief, page 9).

However, the examiner does not agree with Appellant's analysis. As noted in the Office action, FIG. 7B of Chrysos illustrates "reporting" the sampling information or the sampled instruction. Appellant is correct that Chrysos teaches sampling an instruction and storing the sampling information (see, e.g., column 17, lines 26-32). But Chrysos further teaches reading such information after the information is sampled and stored. Chrysos describes, "Step 751 reads the profile record 300 sampled as described above in step 750 ... The record is read when the selected instruction completes" (column 17, lines 36-39). Here, Appellant apparently considers the description of reading the profile record 300 to mean that the stored information is merely "available for access by a software program" (brief, page 9). The examiner submits, however, that when such a software program reads the stored information, the information is in fact "reported" to that software program. Thus, Chrysos teaches "reporting" the sampling information to the software program such as recited in the claims.

Chrysos further teaches a function 755 that filters the stored sampling information (see, e.g., column 17, lines 39-41) and describes, "Step 760 produces a subset of samples based on the function 755" (column 17, lines 52-53; emphasis added). In other words, Chrysos teaches "reporting" a subset of the sampling information based on the function 755. Likewise, Chrysos teaches deriving statistics 790 from the subset of the sampling information (see, e.g., column 17, lines 53-61), including "averages, standard deviations, histograms (distribution), and error bounds of the properties of the sampled instructions" (column 17, lines 54-56). The statistics 790 represent examples of how the sampling information is ultimately "reported" in Chrysos.

Moreover, Chrysos explicitly describes, "Software is informed when the particular selected instruction leaves the pipeline so that the software can read any of the sampled state

information" (column 7, lines 10-12; emphasis added). In other words, information is "reported" to the software when the selected instruction leaves the pipeline. Specifically, Chrysos describes, "When an instruction finishes executing … an interrupt is delivered to higher level software … The software can then process the information present in the profiling registers in a variety of ways" (column 6, lines 62-65). Thus, Chrysos teaches informing or "reporting" the information to the higher level software via an interrupt.

In Chrysos, the "software program" or "higher level software" that reads the sampling information or the sampled instruction represents a "particular thread" such as recited in the claims. Specifically, in Chrysos, the software is executed on a multithreaded processor (see, e.g., column 26, lines 43-46). Chrysos further describes identifying a particular thread of execution (see, e.g., column 11, lines 58-60). Thus, the software includes at least one particular thread of execution. The "events of interest" included in the sampling information (see, e.g., column 15, lines 32-42) are events of interest to the software and therefore to the particular thread of execution. And, as reasoned above, the sampling information or the sampled instruction is "reported" to the software. Thus, Chrysos teaches reporting the sampling information or the sampled instruction "to the particular thread" such as recited in the claims.

Appellant further contends that Kalafatis does not disclose reporting "the sampling information" or "the instruction" to a particular thread (brief, page 9). Appellant contends that in Kalafatis, "information gathered about a sampled instruction is stored and made available to a software program, but not reported to a thread" (brief, page 9).

However, the examiner does not agree with Appellant's conclusion. Appellant acknowledges that "external software" in Kalafatis "may access and sample the contents of event

counters via a program instruction in order to process the stored information" (brief, page 9). Thus, the stored information is "reported" to the external software. Kalafatis describes that an RDMSR instruction is executed to "report" the information to the software (see, e.g., column 6, lines 1-14 and 29-40). As in Chrysos, the software in Kalafatis is executed on a multithreaded processor and therefore includes at least one particular thread of execution (see, e.g., column 2, lines 39-55). Thus, reporting the information to the software includes reporting the information "to the particular thread" such as recited in the claims.

In closing, the examiner notes that Chrysos is directed to sampling instructions in the processor pipeline of a system (see, e.g., abstract). Likewise, Kalafatis is directed to monitoring the performance characteristics of a multithreaded processor (see, e.g., abstract). The examiner respectfully points out that such profiling is performed for a reason. For example, Chrysos describes profiling for the purpose of identifying performance bottlenecks (see, e.g., column 1, lines 29-33). Kalafatis describes profiling for the purpose of improving hardware and software designs (see, e.g., column 1, line 62 to column 2, line 4). The examiner submits that if the profiling information were never "reported" to any particular thread in Chrysos or Kalafatis, as Appellant contends, then there would be no reason to collect and store the information. Indeed, without "reporting" the profiling information, each reference would be unsatisfactory for its intended purpose. Thus, and for the reasons presented above, the examiner respectfully submits that Appellant's arguments are untenable.

Moreover, Appellant is respectfully reminded that the test for obviousness is not that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary

skill in the art. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981). The examiner submits that the combined teachings of Chrysos and Kalafatis would have suggested the claimed subject matter to those of ordinary skill in the art.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Michael J. Yigdall/
Primary Examiner
Art Unit 2192

Conferees:

/Tuan Q. Dam/
Tuan Q. Dam
Supervisory Patent Examiner, Art Unit 2192

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193